ALEX McKENZIE INTERVIEW

This is September 27 in Cambridge, Massachusetts.

The network operation center started out only having anything to do with things that BBN had a responsibility for, which were the IMPs and the circuit center connecting. When the terminal IMP was introduced, ARPA would say to BBN, *"Install a terminal IMP at NASA AMES Research Center."* Okay. We did that. We monitored the terminal IMP. It ran okay. But then people started complaining, *"The Arpanet is no good. The Arpanet never works."* They complained in the press or they'd complain to Larry Roberts. And when I or Frank would hear these statements, we'd say, *"What do you mean it doesn't work?" "Well, I -- you know, Arpa gave me this telephone number to dial up with my modem and whenever I dial it, it doesn't work." "Well, what do you do when you dial?" "Well, you type -- you know, it says something about TIP and you type 'at sign c' and some number, but you know, usually I never get carrier or it hangs up on me."*

So we sent a team of hardware people. It turned out that a lot of these complaints were about the TIP that was at NASA AMES and the modems that were connected to it, and we sent a hardware team out there and we found out that they'd really bought from the low bidder for those modems and the modems were started out being not particularly good and there was nobody at NASA AMES who thought that the modems were their responsibility or ever tested them or looked to see if their *"I'm out of service"* lights were blinking or check to see that the phone lines were really connected to them or anything else. So the network operation center took on the job of at least monitoring the status of all the modems whose numbers were being given out, you know, written on telephone booth walls, practically, by Arpa.

And we wrote some TENEX programs to run an auto-dialer. In these days, that doesn't seem very exciting, but when we did it, the idea of having a computer dial out automatically was practically unheard of. I think we only could find one source of hardware to do that. And we dialed every modem number that we knew about

and we kept statistics on what happened and then we reported into Arpa, or to the site representatives, or if things looked bad enough, we might send our own maintenance engineer out to look at them.

So you began testing the modems --

Every night.

-- every night.

Once a day. Once a day to look for problems before they began to affect too many users, because of course, the naive user was given some number to call and that was the network to him. And we, BBN, and Arpa were going to get bad press if it didn't work, regardless of why it didn't work.

I don't know how many human interest stories or whatever you want, but I remember that there was one modem that had a lot of problems and finally one of the engineers called the number to see if by listening to the noises it made he could figure out what was the matter. Now actually he listened on an extension line -- that's it -- while the auto dialer dialed the number and he heard the phone ring a couple of times and then it was picked up and that was normal, and the auto dialer modem at our end began whistling and some exasperated woman at the other end said, *"Oh, it's you again!"* [pounds table] Smash. So I don't think we'd been dialing that woman for too long, but it was -- we did make an occasional mistake.

With the auto dialer.

Another thing that happened was that Arpa paid for several computers or for accounts on several computers where people could use those accounts for managing electronic mail. There were some in California. There were some in Massachusetts. Probably some in the Washington area. And another kind of complaint that the network doesn't work was because people would try to get to their electronic mail

accounts and the computers would be down and they'd call up the computer center and say, *"When is your computer gonna be up?"* And the computer center operators would say, *"Oh, our computer is fine. It hasn't been down in four days. It must be the network."* And in fact, it was maybe -- I mean, sometimes it was the network. But we began finding there were a lot of cases where the host interface, which had been built by graduate students, had stopped working or the software that ran the network interface had been turned off for some reason and because the site itself was just monitoring -- was the processor processing -- didn't notice. So we added another function for the network operation center which was to monitor the service hosts, the ones that Arpa -- I mean, when we service hosts in this context, I mean a very restricted set. It's the ones where Arpa was funding electronic mail accounts for masses of people and we began monitoring them from the network side and sending traffic into them periodically to see if they were still -- if they still would respond to the traffic, and calling up the site operators and reporting to them that, you know, *"Your network interface appears to be down. Maybe you should check on it."* Frequently they said, *"Well, our whole computer is down."* But sometimes they said, *"Oh, well the computer seems to be running just fine."* And go check into it and discover that the software or the hardware of the network interface was broken.

So our responsibilities grew out of a need for good public relations for the network. I guess that's it. There's one story that I told. I know I told Katie this and maybe you've heard it. If it sounds familiar, stop me. About one of our operators named Dick McDonald going to Hawaii with a sleeping bag and an air mattress. We had a terminal IMP installed at the University of Hawaii and it was going to be used in a big -- it was supposed to be used in a big demonstration for the Navy. I don't remember what month or year, but let's hypothetically say that the big demo was scheduled for June of 1974. That's completely made up. A few months earlier -- in this example let's say it's March of '84 -- Frank Hart went to a meeting of the principal investigator, the Arpa principle investigators, and took unbelievable heat from the principal investigator at the University of Hawaii about what a piece of junk this terminal IMP was and it couldn't ever stay up every day at, you know, 9:00 in the morning. It crashed and it always took us two hours to get it up. How could

Arpa possibly give this demo, which he was going to be at the center of and he was going to take the heat from the Navy, which accounts for his ire, with a machine that wouldn't stay up. We looked at our records and sure enough it did seem to go down a lot and it did seem to be at the beginning of the day in Hawaii and so forth, but there was no -- the diagnostic information we got back every time it crashed was just completely inconclusive. It didn't point in any direction that we could understand.

So finally in this example, I guess it would be in May, as the demonstration grew closer and we still had no clue what was going on, and they were getting ready for a dry run of the demo, I guess, I said, *"Look, I don't know what's the matter. Every time this machine goes down, it takes a long time to get somebody from the University to go to the computer room and give us the readings and stuff like that. If we can't keep the machine up, at least I want to have somebody right there who's ready to get this machine going again and hopefully they'll also be able to give us more diagnostic information about what's going on in the general environment."* So I'm sending one of the network control center operators, Tom McDonald, out there. *"Tom, go buy a sleeping bag and an air mattress at the discount store across the street and then go get on a plane and go there, and I want you to sleep in the computer room. I want you to be there whenever there's not anybody else there. I want you to be there. Call the operation center before even going out for a sandwich."* And Tom went off and bought an air mattress and a sleeping bag and got on the plane and the plane was possibly still on the runway at Logan, certainly not more than half an hour out when the machine went down again and the operation center called up and they called me in pretty quickly. They said, *"You better listen to this conversation."* They said, *"Will you tell me again what happened,"* to the person at the other end, who said, *"Well, you know, it's not like the other times. I dropped my screwdriver into the machine and there were a lot of sparks."* And they said, you know, *"Can you explain to us what you were doing?" "Well, yeah. I'm a graduate student and I'm working on an interface and it needs to have a regulated power supply and I know that the TIP has a regulated power supply at the right voltage, and so every day when I come in, I just tap onto that power supply to get power for my interface, and you know, a lot of times the machine goes down, but you guys always seem to get it back up again, so you*

know, but this time I dropped my screwdriver in and there were a lot of sparks and I'm afraid something really bad has happened." [laughs]

So I called the principal investigator and I said that this had apparently been going on for months, by the admission of the graduate student, and I hoped that he took steps to keep that graduate student from ever going near that machine again, and by that afternoon, he had taken those steps, and by the time Tom McDonald got out there to baby-sit this machine for the week of dry run, there wasn't another crash. So he got to spend a week on the beach at Waikiki during the day so he could be up there at night. But there was never any need of his being there, because we never had another crash from that issue.

The graduate student goes off into the night aimlessly.

Yeah. So fun and games in network operation.

You may not be able to answer this, but let me ask. The maintenance on the IMPs. We were talking with Barker about that and he was explaining that from his point of view the ruggedized cabinet created some problems, because it made parts of the machine difficult to get to.

Uh-huh. That's probably true.

At one point, the liability estimates that had been originally applied to the IMP appeared to be so far off that Arpa thought about canceling the contract with BBN for the IMPs. Does that ring true do you think? The IMPs were breaking about every day for half an hour.

That might right. I know that there was a crises of confidence at Arpa in their estimate of our ability to make the machines run reliably. I have no idea if that was due or related in any way to the machines being in the ruggedized cabinets. I really don't know. I can't comment on that. I do know that there were lots of problems.

That eventually because of the problems in getting decent maintenance, Barker invented a plan to have BBN do the maintenance directly, rather than relying on Honeywell to do it, and I was certainly opposed to doing that, but Barker eventually prevailed on Frank to let him do it, and he managed to get the reliability to go up dramatically. Since he was able to get it to go up dramatically, I really doubt that a first order effect was the ruggedized cabinets. But maybe it was. I mean, I do know, for example, that we were talking earlier about the light switch problem and I can't remember if those lights were really -- I don't think they were actually on the 516. I think the lights that had a problem were on the 316. But Barker basically changed the design of how the lights worked and implemented it as a retrofit to the machines on his own initiative -- his own authority, as it were. So I know that the reliability of the machines as maintained by BBN via the Honeywell field service staff was a real issue at one point. I know that Barker argued for taking over the maintenance and got it fixed. I don't know whether Arpa would have canceled the contracts. I thought that they -- I know that they were very unhappy and I know that they were thinking about taking some other kind of maintenance step. But whether they were really going to throw BBN out entirely, I don't have a recollection of that. Might be true. They were pretty upset. No doubt about that.

Do you recall any specifics about that?

No, it was just that the machines failed a lot. I mean, it's hard to say. It's hard for me to say, especially at this distance, how much of it was things like the Hawaii TIP and things like the perception that this modem bank at AMES was the network being down, and how much of it was really network downs, and then out of those network downs, how much of it was the fault of Honeywell or BBN and how much of it was the fault of the design of the original machine. I can't remember any of the specifics really. But I know that at one point our biggest source of unscheduled outages was site power failures, especially when we began deploying the network into military bases, where the base would shut down its electricity in some building for the weekend, because they were doing something. You know, putting in a new radar. Whatever they were doing, they needed to have the power off and this is a computer

building. Nobody's in there using it. So we'll turn the power off. It didn't occur to them to think that the node in there that nobody was using from that base was carrying traffic across the country. That was a big problem.

No one thought of uninterruptible power supplies?

Well, we thought about a lot of different ways of dealing with issues like that. But we didn't think of them at the beginning. I mean, remember the original contract was to install 14 IMPs in an experimental network, which if it was wildly successful would grow to 19 locations at universities. People weren't anticipating in that original design how quickly -- how badly the community needed this kind of facility, and given how badly they needed it, how quickly they were going to start relying on it. Not as a research network, but as a network to support their research. And Arpa encouraged that, of course. Arpa wanted to have a success and so they encouraged people to think about the network as a utility, as they tried to sign up new uses and new users, without pointing out to them that this was an R&D project.

It was interesting, because I think at one point Barker said that one of the -- there was a geographic focus to some of the trouble and it happened to be Florida where they had bad winds and power would go out more often than in other parts of the country. So that's kind of an unanticipated -- you couldn't anticipate that kind of problem necessarily.

Well, nobody was doing this kind of thing previously. I mean, it really was new. All the networks that existed up to that time were Star networks. So if you had a node in the network, but you weren't processing, turning it off, it wouldn't hurt anybody. And here was a new thing. The university people adapted to that idea. I mean, they didn't always think of it either. But at least they understood the concept if it was pointed out to them. The military bases often it was very, very difficult to get them to understand the concept that if they weren't processing, somebody else still might be using their computer.

You objected to taking over the preventative maintenance.

Yeah.

Why?

Because Barker's office looked as though somebody had backed a dump truck up to it full of scrap paper and emptied it, and I felt that Honeywell might be bad, but Barker was so disorganized he was bound to be worse, and I was going to then take the heat for this network working even worse than it did before.

So it wasn't just a question of competence.

Yeah. Anybody -- I mean, actually I told Frank, *"I absolutely won't accept this. I won't continue on in this job unless Ben at least demonstrates that he can clean up his office."* So he made a determined effort to clean up his office and it was -- at least you could maneuver around in his office after that. If there were stacks of paper, they were stacked. You know, the corners were kind of squared. Just wasn't a mound of stuff.

But he did all right in the end.

He did all right, yes. Yeah, he did.

How large was the maintenance contract on this? I saw a number.

You're going to look at the size of some contract from us to Honeywell or from Arpa to us?

I don't -- it may be just an estimate. This is probably an early estimate -- just the operating cost -- monthly estimate of maintenance at 530 per IMP. I'm looking for a round figure. How much?

[tapes cuts off and back on]

Yeah, well, sure.

[tapes cuts off and back on again]

About the TIP and when the TIP was first conceived and then built and then put into play.

This is another one of those things that was reasonably well documented actually. Probably in documents that we gave to Katie -- in the vast pile of documents we gave her. Certainly one document that I can refer you to was the paper in 19 -- I think it was delivered in 1972, which was about the terminal IMP for the Arpa network. The first paper in this collection of reprints. This was a set of five papers that Larry Roberts had arranged to be given in a session that he shared or organized at the National Computer Conference. I can't remember what the status of the TIP was then, but we must have started writing this paper roughly a year before in the middle of 1971 probably and so, you know, it wasn't very far into the history of the network. I think that the way things worked was that the initial idea was the network would collect together a bunch of campuses. One computer resource per campus. As soon as we started delivering or maybe even before we started delivering the IMPs, it was realized that campuses typically had more than one computer, and so there was the need to go from one interface to IMP and to many, several, and the IMP really was the first local area network for most places. It was the first way that -- even though campuses like MIT had a bunch of different computers sitting right next to each other, they never exchanged any data except by maybe carrying mag tapes or decks of punched cards from one to another. And now if the network will let you exchange data with a dissimilar computer in California, it will also enable you to exchange data with a dissimilar computer in the next room.

Then Arpa began to realize very quickly that if you could get to computer resources only from other places that had computers that implement all the host protocols on, that that was an unnecessarily restricted set and there were lots of people who would like to just be able to dial into something and, of course, modems were just beginning to become popular in those days and dial in and all that kind of thing. So it seemed

natural to event some kind of a miniaturized host computer whose only purpose was to provide access to a geographic base of people who could dial into it. I think that concept must have become pretty powerful by early '71. Maybe even earlier than that, but certainly by early '71.

Now I'm not going to say that another factor isn't that BBN finished working on this contract to build the IMP and wanted to keep the same team of people together feeding at the government trough and wasn't casting about for other things to do, and that might have had something to do with it also. But there were also two competitors to the TIP built. One at the university of Illinois called the ANTS system, Arpa Network Terminal System, and another that was kind of thrown together over a week somewhere in California, by a guy named Dave Retts, that was called the ELF system, also based on a PDP-11. Both of which really had sort of the same purpose as the TIP, but tried to -- by having a whole computer dedicated just to this terminal support function -- tried to overcome of the shortcomings of the TIP which were caused because at least half the processor cycles were going to doing the IMP function instead of the TIP function and the memory was constrained and those kind of things.

But those two others did not in fact get contracts from Arpa.

Oh, they did. Both of them. Arpa supported the development of the ANT System from the beginning and Dave Retts, who put together the ELF System, did it because the company he worked at was waiting and waiting and waiting for the delivery of an ANTS, which kept not coming. I mean, they were waiting for the software. They had the PDP-11 hardware and finally, you know, one of their programmers said, *"Well, I can't do what I'm supposed to be doing until this thing gets here and who knows when it's going to get here, so I'll make one of my own,"* and did. Actually, there were quite a lot of ELF Systems scattered around the network for a while.

I can remember some conversation in the message group later in the '70s which concerned people who were using TIPs, and my sense from reading those messages was that the

people who were using TIPs were called malicious randoms. Not all the people who were using TIPs, but the TIPs changed the fundamental user base. Is there some equation whereby the TIP changed the nature of the people using this network? It broadened it certainly. Yes?

I'm pausing only to try and see if I can recall anything of the times. I think that --

[Tape 1, Side A ends.]
[Tape 1, Side B begins.]

-- at little facilities or they were military officers or, you know, they were not computer science researchers. If they were computer science researchers, by and large they would have had their own computer facilities. So they were other kinds of people -- administrators and bureaucrats and people of that ilk. Having those people on the network changed it certainly, because to begin with that class of people was less tolerant of troubles and was another step toward making the network a utility. The TIP user knew nothing. He had a phone number and a little manual and he was supposed to just use the network, and he didn't care why it didn't work. If it didn't work from his terminal right up to the computer system he was trying to get to, the network was broken. You know, that wasn't a mistake that the people working on Multics research would have made or the people at Englebart's group at SRI that would have said, *"Well, let's see, this computer could be broken or that computer could be broken."* For the typical TIP user, the network was broken.

It is also true that the TIP, because so many of the TIPs were so oriented toward dial-in and the telephone numbers were so widely passed around, because at the beginning there was no log-in or authentication procedure on the TIP. We, BBN, said *"We don't need to have authentication in the TIP because the only thing a TIP is useful for is getting to a host and all the hosts have authentication procedures. So it would be hard for us to do it and it would just be duplicating a function that's done elsewhere, so why should we bother?"* And that did mean that there was a way onto the network for hackers and crackers who then could attack the authentication

systems of other computers on the network. Was that real? I mean, was that a real change? I don't know. I kind of have my doubts. I mean, every host did or could have an authentication system.

The A.I. Lab at MIT prided itself on not requiring any authentication to log-in there. So if you could get onto a TIP and you knew the folklore that you could get onto that MIT A.I. computer without any authentication, then from there you could go trying to attack other people's computers. Is that the fault of the TIP? Is it the fault of the A.I. Lab? Is it nobody's fault? Is it everybody's fault? I mean, I don't really know how to answer that question. People in Silicon Valley who could dial the AMES TIP for a local call and use the network for free to get to the A.I. computer might have been able to conduct a level of hacking that they wouldn't have thought of if they had to pay for a long distance call. That's probably true. But the MIT A.I. computer was there all along and it didn't require any log in or authentication.

It almost may be one of those things where as you just enlarge the number of people involved, you're going to get a greater diversity in the kind of uses.

Yeah, right, yes.

The TIP certainly enlarged the community.

It did enlarge the community. Yes, it did.

I need to ask you about the events that occurred around releasing the IMP code, proprietary code that BBN had written for the IMPs and others wanted.

Right. I'll tell you my perspective. Everybody has a different perspective. Frank absolutely did not wish to release the code. I don't know whether that was his own position or whether BBN, hirer up in BBN, instructed him to take that position. I truly don't know. Frank's argument was that -- Frank's verbal argument, the argument that he made publicly was that the IMPs were not really so protected.

There were lots of ways to get at them. If we gave the code out to a lot of graduate students, then the graduate students, being graduate students, would decide to break into the IMPs and change their code and improve it and futz around with it and the network would stop working and what a terrible idea it was. That's really the only argument I ever heard Frank give or make about it. Was protection of the welfare of the network.

BBN was accused of not releasing the code because it would be valuable to other companies that wanted to build packet switches and sell them to the government, and the code was after all entirely developed with government money and it ought to be in the public domain and therefore BBN should just release it and should stop trying to have a proprietary interest in it. I can believe that the people who said that truly believed that BBN was trying to keep it away from the competitors. I think there's at least one piece of evidence that lends credence to that as a motivation at BBN, even though I never heard Frank say it, and that is that in 1972, I think, a guy named Ralph Alter, who had been at BBN, and another guy whose name was Steve Russell, an engineer who had been in the IMP group, left BBN to form a company called Packet Communications, Incorporated and their plan was for PCI to set up a public packet switching network and offer service to the general public. BBN was pretty upset about that and wanted to do it itself when the time was right, but didn't believe the time was right just then. And I think that there were a lot of people around or there were a number of people around, and Frank might have been among them, who maybe didn't think that the average company -- Deck or IBM or Data General -- could make much use of this code, but that PCI certainly could and that the PCI people were traitors to BBN, kind of.

Walter and Russell had worked on --

Alter, Ralph Alter, A-l-t-e-r, and Steve Russell. Both part of Frank's division. Both had worked on -- Ralph Alter had probably never been a programmer. He isn't like me. Not a programmer or a hardware engineer on the IMP program, but he knew a lot about it. I can't remember exactly what his job was, but he had something to

do with deployment and support and stuff like that.

But to your knowledge, it wasn't part of BBN's business -- long-range business strategy or plan to protect this code so that they could --

No, I didn't say that. I don't know. I find it easy to believe that there were people at BBN who believed that we had the right to that intellectual property and nobody else did. But I don't know for a fact that that's why any particular action was taken, but I can easily believe that it was a strong motivating factor. In fact, BBN did a little bit after the formation of PCI start a company called Telenet and the initial business plan of Telenet was to use Honeywell computers and run the Arpanet IMP code on those computers.

That's what Roberts went off to do.

Steve Levy hired Roberts to be the President of Telenet.

And did Telenet run that code?

I don't think that they ever actually ran that code as a production matter, but they started with it. That was their starting point. So in that sense, it was correct.

Also in that message group -- and this maybe directly related to the IMP code and people trying to get their hands on it. BBN gets called at one point *big bad neighbor* and is seen by at least some people in that community as unwilling to cooperate. Whether that's too strong or not, I'm not sure.

BBN has a longstanding reputation for arrogance. There's absolutely no doubt about it that BBN has a longstanding reputation for arrogance.

What does that stem from? How does it manifest itself?

Well, we were better than other people and we didn't mind telling them so. I mean, most of the other organizations that were involved with the network, especially in the early days, were not trying to make a profit. They were not for-profit organizations and we were. That made us look at a lot of things differently. Where, you know, maybe the primary output of a university was doctoral theses that immediately went into the public domain and the university got its money from tuition or government grants. That wasn't how we got our money to pay our people. So we looked at it differently. I think that there were undoubtedly people who believed that we should have been more collegial, more sharing, and maybe we should have, for that matter. There were also, in our opinion, an awful lot of very half-baked ideas that were being urged by people who didn't have much practical experience that we thought were pretty terrible. When Kleinrock's students wanted to write their own measurement packages and insert them into the IMP without any control exerted by BBN, either on the timing of when they put them in or checking them out beforehand, people like me, responsible for running the Network Control Center, said, *"Hell, no! Can't do it."* The concern wasn't only that they might write code with bugs in it that would crash the machine. The concern was that they might write code that slowed the machine down enough so that it couldn't do its primary job, which was moving packets, and we didn't know how or weren't willing to invest the energy to study the code or random graduate students to assure ourselves that it wouldn't slow the machine down. We just said, *"Tell us what you want and we'll write it."* That pissed a lot of graduate students off. You know? I mean, I can understand why, but I had my job to do too. It was different from their job. So, yeah, we were uncooperative and uncollegial and all those kind of things at times.

Well, it's interesting that you still see this as an experiment. You being BBN.

No, no, absolutely not. We wanted to keep reminding the world that it was an experiment, but we understood perfectly well that our reputation was based on how in this whole community of people who knew we had anything to do with this, that our reputation was based on how well we ran it as a utility. But we were not at all confused about whether it was an experiment or not. It was not an experiment.

Every Tuesday morning during our reserved time when we ran experiments we got angry phone calls. Every Tuesday morning without exception. *"Why is the network down? I worked all night on a proposal and now I'm trying to send it to so and so, and the now the network is down." "Didn't your technical liaison remind you every week - -." "I never heard of anything like this! I want to use the network! Arpa told me this was a reliable thing."* You know, I was not under any illusions that this was an experiment.

You were a utility provider and saw yourself -- yeah.

Yeah. Yes. I, especially as the manager. I mean, I had to remind the internal people of that all the time. *"Well, I have a little patch. I want to just put it out for a couple of hours. I don't need to notify anybody. I've read it over real carefully and it won't break anything."* And I had to keep saying to the internal people, *"Hell, no,"* also. *"You can't do that. I'm not going to let you."*

Tough job.

So I can understand why people would think we were uncooperative. And sometimes we were probably unreasonably uncooperative. I'm not going to debate that either. But we had reasons that the user community often wanted to forget about.

Did that take it out of you personally? That's a fairly combative position to be in.

I'm a fairly combative person, I think. I don't think that was -- yeah, there were times when I went home and I was pretty sick of it and pretty exhausted, but I think that it was okay.

TCP/IP versus OSI. These are Katie's notes to me. The government was torn between letting TCP/IP take over and following the ISO standard. The government was in a bit of a quandary because DOD was pushing TCP/IP while the phone companies wanted to intercommunicate with phone companies overseas.

Yes. All those things are true.

Unfold that a little bit if you could for me, because I don't quite understand all the issues there. This note says that you tried to get those two things closer. That is, TCP/IP and ISO?

ISO, OSI, both.

Okay.

ISO is the International Organization for Standardization and if you write that out in French, its initials are ISO, not IOS, and OSI is Open Systems Interconnection and there were a series of standards for Open Systems Interconnection that were being developed within the ISO. I hardly know where to start. This is a very, very complicated subject.

In 1974, TCP/IP was being invented and the first few countries that were planning to have national packets switching networks were beginning to think about them, and those countries were Canada, the U.S, under Telenet -- the U.S. is always the oddball in this thing, but in all the other countries, it was the National Post and Telecommunications Agency -- Canada, France, the U.K. That's probably it. People like Roberts and his counterparts at the PTT's realized that they needed to have a standard way of connecting -- of running these networks. The equivalent to BBN Report 1822 plus the host protocols that say *how do you connect up to one of these networks?* And these packet networks were going to have TIPs in them, so they couldn't just look at the transport of data. They had to look at the host to host protocol and the Telenet protocol too, because they had to implement that in the analog of the TIP, and the analog of the TIP in their terminology is a PAD, a packet assembler/disassembler.

The TCP/IP argument lead by Cerf and Kahn in this country and by Louis Pruzan and Hubert Zimmerman in France was that it's impossible to make a perfectly

reliable network. That means that if you want perfectly reliable communication you have to make the communication perfectly reliable by mechanisms implemented at the end points of the communication -- in the hosts. Furthermore, we've had a few years of experience with the Arpanet and we see that even those really smart guys -- they must be really smart because they tell us they are all the time, at BBN -- still haven't got it right and there are still lock-ups and routing loops and other stuff like that. It's adding a lot of costs to the Arpanet to try and make it be a reliable network and it's impossible anyhow, even on a theoretical basis, to make a perfectly reliable network. So we should go in the other direction. We should make the networks cheap, easy to implement by anybody, then BBN can have some competitors, and do this reliability stuff in the end points, which we, the host guys, have control over, and we won't have those snobs at BBN telling us what to do anymore. Well, you can understand that argument.

Uh-huh.

But people like Roberts were saying, *"If we're going to use these packet networks that we're about to build, if we're going to manage commercial organizations to use them, we have to make them just as reliable as the Star Networks that people are used to, where they have their terminal and they send stuff to a central host and it gets archived there and backed up and there's transaction numbers and it's really, really reliable. Once it's gone out of your terminal to that Central Star and you've gotten acknowledgement back that the Center Star computer has got it, by God, it's not your responsibility anymore. Somebody else has the responsibility. If we want to compete for that business, we've got to be that reliable. So we want our network to be ultimately superbly reliable."* And they developed the X.25 recommendation and for the PADs, X.3, X.28, and X.29, some other recommendations.

X.3, X.28 and --

And X.29. Those were equivalent to the Telenet and the host to host protocol and the X.25 was equivalent to BBN Report 1822. So they developed those standards and

because they were PTT's, or in Robert's Telenet's case, like a PTT, they got this stuff standardized through the CCITT, the Consultive Committee International Telephone and Telegraph, which is a UN Treaty Organization. That's the meeting place of all the PTT monopolies in the world, and in those days, in the early 70's, the only place there wasn't a PTT monopoly were probably a few places where there were no telephones, plus the United States, where AT&T was effectively the monopoly, but the voting member in CCITT was the US State Department and they had a little committee of AT&T insiders, and Roberts got into that group, Telenet got into that group.

So now there was a conflict. There's packet networking going in two directions. In the research community, it's going the TCP-IP direction of completely unreliable networks. All of the work done in the hosts. And in the X.25 world, it's going in the direction of, *"You're a local phone company. We'll take care of it all for you. Just be good boys. Don't tell us how to run our business. Here's a pat on the head and go have a gumdrop."* That pissed everybody off. Some of the people who were in the TCP/IP camp, although they weren't exactly in the TCP/IP camp, lead by Louis Pruzan of France, with some collaboration, I think, from the unreliable network, put all the work in the host camp in the United States, decided, *"Well, we need another international organization to fight the CCITT. Let's go to the organization that makes computer standards, because they can understand what we're talking about."*

What's the international organization that's makes computer standards? The International Standards Organization, International Organization for Standardization. So the French got introduced into ISO a project to investigate what protocols would be necessary for open computer systems. Open in the sense that they're not proprietary. That is they're not specified by one manufacturer. That project probably started in about 1978. The first meeting was held in Washington and I went to it.

Now International Organization for Standardization, ISO, is an organization which consists of the appointed delegates of national standards organizations. In the United

States in the field of computers, the national standards organization that's accredited to ISO is ANSI, the American National Standards Institute. Members of ANSI are people who pony up a fairly large amount of money, first to be on their mailing lists and I mean, it's necessary because they generate and distribute huge volumes of documents, so you have to pay something for that, and second, are willing to go to one meeting every month or two months, sort of forever, to make standards. That's quite a contrast with the network working group or the Internet working group which is mostly like graduate students who ponied up a few ideas and their plane fare to get to the meeting. The open system's interconnection endeavor in ISO which had been started really by the radicals, the INWG network working group kind of people, before very long, was captured by the kind of companies that had the money and energy to put into ANSI membership and membership in the British Standards Institute and the French Association for Standardization and the German Standardization, companies like Deck and IBM and NCR and Philips and Siemans and Bohl and Hewlett Packard and so forth. These people whose professional careers for a period of several years was going to the standards meeting, because the more influence you --

[Tape 1, Side B ends.]
[Tape 2, Side A begins.]

-- quickly the radicals in TCP/IP, the proponents of TCP/IP had lost influence, if they ever thought they were going to have any, to the representatives of big companies. Well, big companies all thought more like the telephone companies in CCITT than they did like anybody else. They figured if they're going to endorse a network and implement it in their products it's got to be pretty reliable and they were all used to centralization and all of that. So after many years of effort -- well, that's jumping ahead. It became clear quite early on that the ISO was going to adopt a set of open system standards which took as a given X.25 as the network standard and build up from there in the same way which the TCP/IP people felt was inappropriate. In the United States, the National Bureau of Standards, now it's called NIST, but in those days it was called NBS, represents the needs of the

government in computer standardization in ANSI. The Defense Department said the whole idea of all this packet switching stuff was to have this reliable network where you break some one piece and it doesn't all fail, and we've become convinced over the years, especially because Arpa's been pushing on us so hard and we don't really have any other good source of information, that a really distributed network technology, where the networks are cheap and don't have to have all this reliability mechanism, are the way to go. So NBS please go gather up the opinions from all the other departments in the government and then go get the ISO guys to adopt TCP.

NBS hired BBN. I'm sure this was a competitive procurement and I wasn't in on the beginning of it. I don't know who we were competing against. To provide them (them being NBS) with technical expertise.

Do you remember the year?

I think the contract started in 1980, but I'm not positive of that. '79. I bet it was '79. Government fiscal year '79. To provide them with technical expertise in national and international standards deliberations, in evaluating proposals, and helping them to promote the interests of the U.S. government. So BBN got to go to ANSI meetings and try to establish our credentials there. Now, in ANSI, we couldn't sit in the ANSI meetings as representatives of NBS. We had to sit there as representatives of BBN.

Were you doing this?

I ended up taking it over in 1981, I guess.

Who was doing it before you?

It was the manager of that project. A guy named Mike Wingfield. Mike Wingfield ties back into this because Mike Wingfield built the host interface for the UCLA Sigma 7, which was the first host on the first IMP in the Arpanet. He was a

graduate student at that time. So I and a team of people at BBN -- there were probably about six people on the team -- went to ANSI meetings and to ISO meetings to represent the point of view that what the U.S. government needed was a protocol that was like TCP/IP. Well, TCP/IP isn't perfect. You know, you look at it and it can always be improved. The ISO for political reasons has the general philosophy of never adopting anything as it's submitted and the reason is that most of the people who come to ISO as national experts are from some company which has a nationality and has a product to push, and if IBM gets to walk in and say, "*Here's the design of a communication protocol. Let's adopt this as the international standard.*" That's bad for everybody except IBM. So first ANSI will never adopt anything as submitted so as to not favor one company over another and then ISO will almost never adopt anything as submitted so as to not favor one country over another.

So TCP/IP were introduced into ISO, via ANSI, as another protocol at the same level as X.25 and the transport protocol built on top of X.25, as other alternatives and in sort of co-equal standard status, and then in its usual way ISO debated them at length and modified them a fair amount and then adopted them. There is a protocol quite similar to IP called the Connectionless Network Layer Protocol and a protocol quite similar to TCP called Transport Protocol Version 4, Variant 4, or something like that, typically called TP-4. There's TP-0 which is sort of no protocol at all. It's designed to fit exactly over X.25, which tries to be perfectly reliable, so you don't need any protocol over it, and there's TP-4, which is designed to fit over the Connectionless Network Layer Protocol, CLNP, Connectionless Network Protocol. Then there's some variance in between TP's 1 through 3, which I don't think anybody pays any attention to. They were there to solve other political battles, not this political battle.

So NBS was successful. They got ISO to adopt TCP and IP in essence. The TCP/IP community was not the least bit mollified. They didn't adopt it exactly. Furthermore, it wasn't the standard. It was one of five standards at the transport layer or one of two or three standards at the network -- I guess only two -- connectionless and connection oriented -- at the network layer. Furthermore, ISO

had taken so much time debating it. No, that's not right. If there was a real clear push at the time that ISO adopted these things to switch to them, it might have worked. There might have been a switch. That big network explosion hadn't -- the big Internet explosion hadn't yet happened. But because there were five different transport protocols and at least two different layer protocols and they were radically different and they couldn't interwork with each other, and not only that, in the transport layer protocol there were dozens of options and in the next layer up, the session layer, there were hundreds of options and they were all totally incompatible. Nobody rushed out to implement. They were waiting for somebody else to do it first and see what gained acceptance in the marketplace and they'd implement. Well, while people were waiting to see what gained acceptance in the marketplace, TCP/IP gained acceptance in the marketplace and now almost nobody talks about OSI anymore. I mean, not that I notice. So it was all kind of strange.

That story of standards arising from de facto situations is fairly common.

It is. It is.

Just seems to be the way these things get done, rather than imposed. It doesn't always result in the perfect standard either. Does it?

No. No.

It's sort of get there first.

Well, you know, a thing that the group responsible for developing protocol standards for the Internet is fond of pointing out is that almost by definition almost no ISO standard gets adopted if there's an implementation of it. Because that would give somebody an advantage. And in the Internet, almost no standard is ever adopted until there are two independent implementations of it proven to be able to work with each other. That's a big philosophical difference. So far the race has gone to the Internet approach where it's been in competition with the ISO approach. There's

not so many places. I mean, ISO has 10,000 standards or something like that. They cover all kinds of things. A standard for wine glasses, a standard for condoms. They have standards for everything. In this particular area, it hasn't worked out so well. But ISO is still a strong and viable organization.

Sure. Oh, sure. The people inventing Telenet in '79, Cerf, Steve -- [tape cuts off and back on again] -- NCP and Tunnel. What is format of messages? How much work does each -- [tape cuts off and back on again] What was your job interfacing with -- [tape cuts off and back on again] --

BBN, yes.

Okay.

Right. I was the outside guy. I've told Katie this, but maybe it hasn't gotten to you. What happened is that I took a leave of absence from BBN in 1970 in May, a six month leave of absence, and my wife and I, who had no children and no property, went to Europe and bought a Volkswagen Beetle and a tent and went camping for six months and touring around and it was wonderful. When I came back, it was November of 1970. There were an increasing -- by November of 1970, we must have had 10 nodes installed, something like that, 12 maybe. There were a lot of people from the host organizations who hadn't really expected this thing to ever work or ever really come to their location and they were under a lot of pressure to implement their host interface. They were under a lot of pressure from Larry to use the network for something. Every new organization had all the same questions. In the meantime, even though we had all that number of IMPs installed, there were still plenty of problems with the hardware and the software. We talked a little earlier about the hardware reliability. There was software reliability. It was still an experiment in 1970. In 1970, it was still an experiment. It was not a utility.

So I came into the building with nothing to do. I mean, I hadn't been doing anything for BBN for six months. They learned to get along without me. Frank

knew I was pretty smart and he said, *"Look, what I'd like you to do for a while is to be the front man for the IMP group, so that instead of every one of these organizations asking the implementers the same questions and having the same arguments over and over again and distracting them from what they're trying to do, which is really make the network work well, you can ask them once and I have confidence that you'll remember the answer and won't come back and bug them again, and then you can go answer the questions."* So I became the network generalist and the outside interface. Not on really deep technical things. I mean, Walden and Crowther and Ornstein and Kahn still talked to people about deep things. But just in terms of, you know, general knowledge. *How does the IMP interface work? What does this mean?*

Frequently asked questions.

Yes, right. Well, frequently asked and then some not so frequently asked questions. And that turned into also representing the IMP group at the network working group meetings. For example, in the design of the Telenet protocol, I was representing the TIP. The TIP had extremely limited memory. It had to support an extremely large band width for those days of terminals with only half of a machine that wasn't very powerful, because the other half was being the IMP. So I had to argue on the side of simplicity, and let's not make it too complicated, and easy commands, and not too much state to remember, and not too many fancy things. While other people who represented other points of view were representing their points of view.

In the design of the host protocol, in those days, we hadn't thought very much about layering and all of that. People generally believed, *"Well, if the IMP is already doing something, there's no point in the host doing it too. So let's bum all we can off of what ever the IMP is doing."* And so I was there to explain what the IMP was doing, to carry requests for the IMP doing slightly different things back to the group, you know, to convey the answers back. *"Yeah, we can do this." "No, we can't do that."* So that was my role in the host protocol design. Besides that, I'm not an inventor. I mean, it's not my character, my forte to invent. But I'm a good codifier. I was always fond in Junior High when I learned about (Hammah Rabbi?) who codified

the law. That's the thing that I'm good at. So since I was a BBN employee, my job was -- this was my job. I was not also trying to write a thesis or teach graduate students or anything. I wrote a lot of the protocol documents and, of course, as we know, the person who wields the pen at the end gets to decide all the ambiguous cases any way they want. So BBN was happy for me to have that job. I was happy to have it. I was good at it. I could write clearly. Because I'd been in the technical arguments, I understood how we had gotten to the points we got to. So I didn't write them down in some fallacious way that had logical flaws in it. So I documented a lot of the protocols. And because I could do that and because I was not a threat to their inventiveness, the inventive guys liked to have me on their committees. I would do all the grunt work. I wouldn't challenge their ideas except when they were really flawed. You know, I had established my own credibility that way. So I did that for a few years.

When you say you were in a position of arguing for simplicity, ease of use, spareness, that kind of general --

Ease of use, not so much. Each of implementation was what I -- can you implement it in a small amount of code, in a small memory, without storing too many variables up for a long time? That was the point of view that I had to represent.

Does that mean that others were coming at you with complicated requests?

There was the idea that for every Telenet connection somebody could generate a mapping table for all the character codes in this machine and all the character codes on this terminal and the character code mapping could be perhaps downloaded into the machine that was driving the terminal. Well, that was the TIP. We didn't have room for 64 character mapping tables with 128 or 256 entries in each one. Hell, no. Couldn't do it. One of the biggest debates in the whole Telenet design and it seems bizarre. Well, it'll seem bizarre when I tell you about it. It may be one of the last things I get to tell you today. It's the debate over how to terminate a line of text. In the Tenex system, the Deck system, and in other systems too, there are two keys,

a carriage return key and a line feed key, and if you wanted to go to the beginning of the next line, you sent carriage return line feed. If you wanted to go back to the beginning of the line, let's say to underline the line, you just typed *carriage return* and then typed underlines, and you overstruck the previous character positions. In the IBM systems, on their terminals, the 2741 terminal, there was a new line key. The new line combined the functions of carriage return and line feed into one key. The IBM system's file structure was mostly records structured and so a new line was like the end of a record. If by some chance you wanted to overstrike, you sent backspace.

You got to the end and t-t-t-t.

Okay, yeah, right, and you backspaced. The Tenex and Multics systems -- well, no. Multics was yet another case. The Tenex systems were very character oriented and not record oriented. Their file structure was character oriented and so to them there shouldn't be -- to the Tenex people, a line wasn't a record. A line was just some characters in the middle of a big bunch of characters. The storage in the file systems of the machine was different. The interpretation was different. Okay. Now, you're communicating between these two systems. How do you signify the end of a line? The endless debates about it. Furthermore, because the network was asynchronous and because message boundaries had no meaning, you could send a carriage return now and a little while later you could send the next character. So what was the machine supposed to do with the carriage return? Well, some people said you should count the number of characters that you've already printed in the line and down count when you do backspaces and up count when you do forward spaces and count eight when you do tabs unless there's been a tab setting command issue, in which case, you'd count the number of tabs, and if it seems like the end of the line, turn a carriage return into a new line, and otherwise don't. Well, TIP couldn't do that. It's a lot of programming for handling the end of the line. The solution that we all ended up compromising on was if you meant new line, you'd send carriage return line feed and if you meant anything else, you'd send carriage return null. So we only needed one bit of state. It was the last thing that I process, the carriage return, yes

or no, and if it was then when you got a line feed or a null, then you knew what to do. If it wasn't a carriage return, then a null meant throw it away, and a line feed meant stay where you are horizontally on the line, but go down one line, because maybe I'm typing a table and you ought to go down a line.

So it was those kind of incompatibilities at the very foundation of -- I mean, this really had to do with the record structure that was adopted on an IBM system and not adopted in the Tenex systems. There was no records structure. There was only a file structure -- a recordless file structure. The IBM guys couldn't understand that. The Tenex guys couldn't understand why you'd want to call a big block of text records. Neither of them could talk to the other ones. Of course, maybe in some previous job or graduate school or something, they'd had a job where the other convention was used, but maybe not. In all of these debates, my job was to not try and argue for one or another based on anything other than what would be easy to implement, and in the design of the Telenet protocol, my job was to try and make the implementation be as small and compact as possible. That was my assignment.

So you would always lean toward that kind of --

I had to look at -- I had to try and understand, *"Okay, I've got a program. Here are some competing ideas. How would I program this? How would I program this? Which one requires less space? Which one requires less cycles?"* That was my assignment.

That's a very specific kind of exercise or rule applied to these problems. Not an easy problem necessarily.

Yeah, right.

But in the end result, would you characterize that as a command decision that this is the way it shall be?

Oh, no. I just had to be persuasive. Sometimes, every now and then I could say, *"Oh, please, don't do that. It will be so hard for the TIP."* But you can only use the same argument once or twice in the course of a year. Most of the time I had to try and line up support on some other basis. I mean, it's all negotiating. It's just groups with different points of view. If I was doing my job well most of the time, people should not have realized that my primary concern was implementation.

People should not have realized it?

They should not. The rest of the group I was negotiating with, arguing with, designing with, should not -- I should not have presented it in such a way that made them understand that that was my concern, because then they'd ignore it. I mean, why should they care? Why should they all contort their systems to maybe something easy for BBN?

I see. So you actual motive had to be concealed. Not concealed, but --

I mean, if anybody thought about it, they could understand. But it couldn't be blatant. Concealed is too strong. But blatant is what it couldn't be.

Were you successful?

Mostly. I mean, you can't win everything.

Do you remember any specific instances in which you sort of walked away and said, *"Boy, did I have to cave there,"* or make a big concession?

I don't remember any. No. I mean, I'm sure there were times. But I don't remember. It made it easy that this whole group were friendly collegial people. In ISO, it was different. In ISO, I ended up permanently furious at a few people for just being so dogmatic and failing to be willing to make any effort to understand another point of view. In the network working group, everybody tried to understand

what other people were saying. It was a friendly -- the group -- we could argue all day and then we could go out for beers and seafood at night, or beer and pizza, or whatever, and it wasn't --

Who were the big bosses in the network group?

Postell and Crocker, of course; Harslem and Hefner, all they were in it; me; a guy named Bob Braden from UCLA Campus Computer Network. He was the IBM representative. Mike Pavliski from Multics.

And if that core group sort of formed a mass or decision or were moving in one direction --

If Pavliski and Crocker and me and Postell -- [tape cuts off and on] -- Braden(?) could agree on something, it would pass.

Did you then meet more often with those guys than with the group as a whole?

No, I don't think so. No, I don't think so. I think everything was pretty much out in the open. Again, unlike ISO and ANSI, there was not back room politicking. I don't know of anything that the network working group did that I'm aware of it doing that was done on other than real open technical discussion. I can't say that Crocker and Postell didn't caucus when they were graduate students together at UCLA, or even that they didn't go across to see Braden, who was -- actually, they probably never would have gone across the campus to see Braden. They thought Braden was the enemy. Braden and Pavliski were more natural allies than anybody else.

[End of Tape 2, Side A.]